



Avalara AvaTax for

AX 2012 R3 CU13 Retail

Code Change Document - EPOS customisations

Version 1.1

Revision date: 05/11/18

Product release #: Avalara AvaTax for AX 2012 R3 CU13 Retail Version AX 6.3.6000.151.01.00



Avalara may have patents, patent applications, trademarks, copyrights, or other intellectual property rights governing the subject matter in this document. Except as expressly provided in any written license agreement from Avalara, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2013 Avalara, Inc. All rights reserved.

Avalara, AvaTax, AvaTax Calc, AvaTax Returns, AvaTax Certs, AvaTax Local, AvaLocal, AvaTax POS,

Prerequisite:

Destination folder should contain Microsoft.Dynamics.Retail.Pos.Contracts.dll with relevant version

Installation Steps:

Add reference of following two files in VS project:

- 1) **Avalara.Ax.Retail.dll**
- 2) **Avalara.AvaTax.Adapter.dll**
- 3) **Code changes in shared projects**

AvaTax Changed the following list of Projects in existing Retail SDK projects

Project/Form Name	Methods	Code Changes
Tax.cproj	Project → Properties → Setting	<p>Set value of ProviderAssemblies as</p> <p>*</p> <p>Extensions\Avalara.Ax.Retail.dll</p> <p>Set value of ProviderClassNames as</p> <p>*</p> <p>Avalara.Ax.Retail.AvaTaxProvider</p>
BlankOperations.cproj		<p>Please add the following reference to the project</p> <p>Avalara.AvaTax.Adapter.dll Avalara.Ax.Retail.dll</p>
BlankOperations.cproj	BlankOperations.cs	<p>Search for the following</p> <p>using Microsoft.Dynamics.Retail.Pos.Contracts.Services;</p> <p>Add the following lines after that.</p> <p>using System; using Avalara.Ax.Retail.Util; using Avalara.Ax.Retail.Diagnostics;</p>
BlankOperations.cproj	BlankOperations.cs/ BlankOperations	<p>Search for the following lines of code</p> <p>StringBuilder comment = new StringBuilder(128);</p> <p>comment.AppendFormat(ApplicationLocalizer.Language.Translate(50700), operationInfo.OperationId); comment.AppendLine();</p> <p>comment.AppendFormat(ApplicationLocalizer.Language.Translate(50701), operationInfo.Parameter); comment.AppendLine();</p>

```

comment.Append(ApplicationLocalizer.Language.Translate(50702));

        using (LSRetailPosis.POSProcesses.frmMessage dialog = new
LSRetailPosis.POSProcesses.frmMessage(comment.ToString(),
MessageBoxButtons.OK, MessageBoxIcon.Error))
        {

LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog)
;
        }

```

And **replace** with the following lines of code

```

if (operationInfo.OperationId == "9999")
    {

        //LSRetailPosis.POSProcesses.WinControls.StringPad

        switch (operationInfo.Parameter.ToUpper())
        {

            case "ECMS":
                //Sets exemption number, scope of this number will be
limited for current transaction.
                using (LSRetailPosis.POSProcesses.frmInputNumpad
inputDialog = new LSRetailPosis.POSProcesses.frmInputNumpad())
                {
                    inputDialog.EntryTypes =
Contracts.UI.NumpadEntryTypes.None;
                    inputDialog.MaxNumberOfDigits = 25;
                    inputDialog.PromptText =
ApplicationLocalizer.Language.Translate("Enter Exemption no:", -1);

                    if
(Istring.IsNullOrEmpty(TaxSvcHelper.exemptionNumber))
                    {
                        inputDialog.InputText =
TaxSvcHelper.exemptionNumber;
                    }

LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(inputD
ialog);

                    if (inputDialog.DialogResult == DialogResult.OK)
                    {
                        if (inputDialog.InputText.Length > 25) //Client side
validation for ECMS
                        {

//InternalApplication.Services.Dialog.ShowMessage(
// "ExemptionNo lenght must be between 0 and 25
characters.", MessageBoxButtons.OK,

```

```

        // MessageBoxIcon.Error);
        using (LSRetailPosis.POSProcesses.frmMessage
dialog = new LSRetailPosis.POSProcesses.frmMessage("ExemptionNo
length must be between 0 and 25 characters.", MessageBoxButtons.OK,
MessageBoxIcon.Error))
        {
LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog)
;
        }
        }
        else
        {
            TaxSvcHelper.exemptionNumber =
inputDialog.InputText;
        }
    }
    break;
    case "EUCODE":
        //Sets entity use code, scope of this number will be limited
for current transaction.
        using (LSRetailPosis.POSProcesses.frmInputNumpad
inputDialog = new LSRetailPosis.POSProcesses.frmInputNumpad())
        {
            inputDialog.EntryTypes =
Contracts.UI.NumpadEntryTypes.None;
            inputDialog.MaxNumberOfDigits = 40;
            inputDialog.PromptText =
ApplicationLocalizer.Language.Translate("Enter Entity Use Code:", -2);

            if (!string.IsNullOrEmpty(TaxSvcHelper.entityUseCode))
            {
                inputDialog.InputText = TaxSvcHelper.entityUseCode;
            }

LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(inputD
ialog);

            if (inputDialog.DialogResult == DialogResult.OK)
            {
                if (inputDialog.InputText.Length > 40) //Client side
validation for Entity use code
                {

//InternalApplication.Services.Dialog.ShowMessage(
// "EntityUseCode length must be between 0 and
25 characters.", MessageBoxButtons.OK,
// MessageBoxIcon.Error);
                using (LSRetailPosis.POSProcesses.frmMessage
dialog = new LSRetailPosis.POSProcesses.frmMessage("EntityUseCode
length must be between 0 and 25 characters.", MessageBoxButtons.OK,
MessageBoxIcon.Error))
                {

```

		<pre> LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog) ; } } else { TaxSvcHelper.entityUseCode = inputDialog.InputText; } } break; case "DIAGNOSTICS": Diagnostics diagnostics = new Diagnostics(); diagnostics.ShowDialog(); break; default: break; } } else { StringBuilder comment = new StringBuilder(128); comment.AppendFormat(ApplicationLocalizer.Language.Translate(50700) , operationInfo.OperationId); comment.AppendLine(); comment.AppendFormat(ApplicationLocalizer.Language.Translate(50701) , operationInfo.Parameter); comment.AppendLine(); comment.Append(ApplicationLocalizer.Language.Translate(50702)); using (LSRetailPosis.POSProcesses.frmMessage dialog = new LSRetailPosis.POSProcesses.frmMessage(comment.ToString(), MessageBoxButtons.OK, MessageBoxIcon.Error)) { LSRetailPosis.POSProcesses.POSFormsManager.ShowPOSForm(dialog) ; } } } </pre>
Customer.cproj		<p>Please add the following reference to the project</p> <pre> Avalara.AvaTax.Adapter.dll Avalara.Ax.Retail.dll </pre>

Customer.cproj	frmNewCustomer.cs/ SaveCustomer	<p>Please search for the following line</p> <pre>using DM = Microsoft.Dynamics.Retail.Pos.DataManager;</pre> <p>Add the following lines after that.</p> <pre>using System.Data.SqlClient; using Avalara.AvaTax.Adapter.AddressService; using Avalara.Ax.Retail; using Avalara.Ax.Retail.Util; //using LSRetailPosis.Transaction; //using Microsoft.Dynamics.Retail.Pos.Contracts; using AvaAddress = Avalara.AvaTax.Adapter.AddressService.Address; using System.Collections.ObjectModel;</pre>
Customer.cproj	frmNewCustomer.cs/ SaveCustomer	<p>Please search for the following line</p> <p>Note : There could be multiple occurrence present in the function with the below string . Please consider the 1st occurrence.</p> <pre>IList<Int64> entityKeys = new List<Int64>();</pre> <p>Add the following lines after that.</p> <pre>//AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && TaxSvcHelper.countryListForValidation.Contains(Convert.ToString (this.addressViewModel.Address.Country))) { AddressSvcWrapper addressSvc = new AddressSvcWrapper(); ValidateRequest validateRequest = new ValidateRequest(); validateRequest.Address = new AvaAddress(); validateRequest.Address.Line1 = this.addressViewModel.Address.StreetName; validateRequest.Address.City = this.addressViewModel.Address.City; validateRequest.Address.Region = this.addressViewModel.Address.State; validateRequest.Address.PostalCode = this.addressViewModel.Address.PostalCode; validateRequest.Address.Country = this.addressViewModel.Address.Country; ValidationResult validateResult = addressSvc.Validate(validateRequest); DialogResult ValidateAddressPrompt; if (validateResult.ResultCode == Avalara.AvaTax.Adapter.SeverityLevel.Success) { if (validateResult.Addresses.Count > 0) {</pre>

```

string validateAddressMsg =
    string.Format("Address
Validated by AVATAX{0}{1} {2} {3} {4} {5} {6} {7}",
Environment.NewLine, validateResult.Addresses[0].Line1,
validateResult.Addresses[0].City,
validateResult.Addresses[0].Region,
validateResult.Addresses[0].PostalCode,
validateResult.Addresses[0].Country, Environment.NewLine,
"DO you want
to update?");

ValidateAddressPrompt =
Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowDialog.ShowM
essage(
    validateAddressMsg,
    MessageBoxButtons.
        YesNo,
    MessageBoxIcon.
        Question);

if (ValidateAddressPrompt ==
DialogResult.Yes)
{
this.addressViewModel.Address.StreetName =
validateResult.Addresses[0].Line1;

this.addressViewModel.Address.City =
validateResult.Addresses[0].City;

this.addressViewModel.Address.State =
validateResult.Addresses[0].Region;

this.addressViewModel.Address.PostalCode =
validateResult.Addresses[0].PostalCode;

//TODO: need to write
mapping code between Avalara country code and AX country code.

//this.addressViewModel.Address.Country =
validateResult.Addresses[0].Country;

}

}
else
{
string validateAddressErrMsg =
string.Format("{0} {1} {2}", Environment.NewLine,
validateResult.Messages[0].Name,

```

		<pre> validateResult.Messages[0].Details); Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM essage(validateAddressErrMsg, MessageBoxButtons.OK, MessageBoxIcon.Information); } System.Data.SqlClient.SqlConnection sqlDataConnection = Pos.Customer.Customer.InternalApplication.Settings.Database.Con nection; string sqlString = "select [RETAILTAXGROUPID] from [ax].[AVA_CONFIGURATION]"; System.Data.SqlClient.SqlCommand sqlCommand = new SqlCommand(sqlString, sqlDataConnection); sqlDataConnection.Open(); System.Data.SqlClient.SqlDataReader sqlReader = sqlCommand.ExecuteReader(); if (sqlReader.HasRows) { sqlReader.Read(); strAvaTaxGroup = sqlReader[0].ToString(); } } //AVA-</pre>
Customer.cproj	frmNewCustomer.cs/ SaveCustomer	<p>Please search for the following line Note : There could be multiple occurrence present in the function with the below string . Please consider the 1st occurrence.</p> <pre> if (createdAx) {</pre> <p>Add the following lines after that.</p> <pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && !Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalcu lation) { ReadOnlyCollection<object> carr; carr = Pos.Customer.Customer.InternalApplication.TransactionServices.I nvokeExtension("ava_updateCustomer", tempCustomer.CustomerId); } //AVA-</pre>

Customer.cproj	frmNewCustomer.cs/ SaveCustomer	<p>Please search for the following line</p> <pre>createdLocal = customerDataManager.SaveTransactionalCustomer((tempCustomer as LSRetailPosis.Transaction.Customer), entityKeys);</pre> <p>Add the following lines after that.</p> <pre>//AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && !Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalcu lation) { this.viewModel.Customer.SalesTaxGroup = strAvaTaxGroup; this.addressViewModel.Address.SalesTaxGroup = strAvaTaxGroup; } //AVA-</pre>
Customer.cproj	ViewModel\AddressVi ewModel.cs	<p>Please search for the following line</p> <pre>using DM = Microsoft.Dynamics.Retail.Pos.DataManager;</pre> <p>Add the following lines after that.</p> <pre>using System.Data.SqlClient; using AvaAddressService = Avalara.AvaTax.Adapter.AddressService; using Microsoft.Dynamics.Retail.Pos.Contracts; using AvaAddress = Avalara.AvaTax.Adapter.AddressService.Address; using System.Collections.ObjectModel; using Avalara.Ax.Retail.Util; using Avalara.Ax.Retail;</pre>
Customer.cproj	ViewModel\AddressVi ewModel.cs\ SaveShippingAddress	<p>Please search for the following line</p> <pre>IAddress tempAddress = this.Address;</pre> <p>Add the following lines after that.</p> <pre>//AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableAddress Validation && TaxSvcHelper.countryListForValidation.Contains(Convert.ToString (this.Address.Country))) { try { AddressSvcWrapper addressSvc = new AddressSvcWrapper(); AvaAddressService.ValidateRequest validateRequest = new AvaAddressService.ValidateRequest(); validateRequest.Address = new Avalara.AvaTax.Adapter.AddressService.Address(); validateRequest.Address.Line1 =</pre>

		<pre> this.Address.StreetName; validateRequest.Address.City = this.Address.City; validateRequest.Address.Region = this.Address.State; validateRequest.Address.PostalCode = this.Address.PostalCode; validateRequest.Address.Country = this.Address.Country; AvaAddressService.ValidateResult validateResult = addressSvc.Validate(validateRequest); DialogResult ValidateAddressPrompt; if (validateResult.ResultCode == Avalara.AvaTax.Adapter.SeverityLevel.Success) { if (validateResult.Addresses.Count > 0) { string validateAddressMsg = string.Format("Address Validated by AVATAX{0}{1} {2} {3} {4} {5} {6} {7}", Environment.NewLine, validateResult.Addresses[0].Line1, validateResult.Addresses[0].City, validateResult.Addresses[0].Region, validateResult.Addresses[0].PostalCode, validateResult.Addresses[0].Country, Environment.NewLine, "DO you want to update?"); ValidateAddressPrompt = Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM essage(validateAddressMsg, MessageBoxButtons. YesNo, MessageBoxIcon. Question); if (ValidateAddressPrompt == DialogResult.Yes) { this.Address.StreetName = validateResult.Addresses[0].Line1; this.Address.City = validateResult.Addresses[0].City; this.Address.State = validateResult.Addresses[0].Region; this.Address.PostalCode = validateResult.Addresses[0].PostalCode; } </pre>
--	--	---

```

        }
    }
    else
    {
        string validateAddressErrMsg =
string.Format("{0} {1} {2}", Environment.NewLine,
validateResult.Messages[0].Name,
validateResult.Messages[0].Details);

Pos.Customer.Customer.InternalApplication.Services.Dialog.ShowM
essage(
                validateAddressErrMsg,
MessageBoxButtons.OK, MessageBoxIcon.Information);
    }

    if
(string.IsNullOrEmpty(this.Address.SalesTaxGroup))
    {
        System.Data.SqlClient.SqlConnection
sqlDataConnection =
Pos.Customer.Customer.InternalApplication.Settings.Database.Con
nection;

        string sqlString = "select
[RETAILTAXGROUPID] from [AX].[AVA_CONFIGURATION]";
        System.Data.SqlClient.SqlCommand
sqlCommand = new SqlCommand(sqlString, sqlDataConnection);
        sqlDataConnection.Open();
        System.Data.SqlClient.SqlDataReader
sqlReader = sqlCommand.ExecuteReader();
        string strAvaTaxGroup =
string.Empty;

        if (sqlReader.HasRows)
        {
            sqlReader.Read();
            strAvaTaxGroup =
sqlReader[0].ToString();

            this.Address.SalesTaxGroup =
strAvaTaxGroup;
        }
    }
}
catch (Exception ex)
{
    TaxSvcHelper.retailTransactionTrace.AddToTrace("Inside
SaveShippingAddress, error : " + ex.Message);
}
}
//AVA-

```

TransactionTriggers.cproj		<p>Please add the following reference to the project</p> <p>Avalara.AvaTax.Adapter.dll Avalara.Ax.Retail.dll Microsoft.Dynamics.Retail.Pos.SystemCore.dll</p>
TransactionTriggers.cproj	TransactionTriggers.cs	<p>Please search for the following line</p> <p>using Microsoft.Dynamics.Retail.Pos.Contracts.Triggers;</p> <p>Add the following lines after that.</p> <pre>using System.ComponentModel.Composition; using System.Data.SqlClient; using LSRetailPosis; using Microsoft.Dynamics.Retail.Pos.Contracts.DataEntity; using Microsoft.Dynamics.Retail.Pos.Contracts.Triggers; using Avalara.Ax.Retail.Util; using System; using System.Collections.ObjectModel; using LSRetailPosis.Transaction; using Microsoft.Dynamics.Retail.Pos.SystemCore;</pre>
TransactionTriggers.cproj	TransactionTriggers.cs/ BeginTransaction	<p>Please search for the following line</p> <p>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.BeginTransaction", "When starting the transaction...", LSRetailPosis.LogTraceLevel.Trace);</p> <p>Add the following lines after that</p> <pre>//AVA+ TaxSvcHelper.isReturnTran = false; //AVA-</pre>
TransactionTriggers.cproj	TransactionTriggers.cs/ PreEndTransaction	<p>Please search for the following line</p> <p>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.PreEndTransaction", "When concluding the transaction, prior to printing and saving...", LSRetailPosis.LogTraceLevel.Trace);</p> <p>Add the following lines after that</p> <pre>//AVA++ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalculation) { if (posTransaction is RetailTransaction) { RetailTransaction retailTransaction =</pre>

		<pre>(RetailTransaction)posTransaction; TaxSvcHelper.SetInvoiceComment(retailTransaction); } } //AVA-</pre>
TransactionTriggers.cproj	TransactionTriggers.cs/ PostEndTransaction	<p>Please search for the following line</p> <pre>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.PostEndTransaction", "When concluding the transaction, after printing and saving", LSRetailPosis.LogTraceLevel.Trace);</pre> <p>Add the following lines after that</p> <pre>//AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalculation) { if (posTransaction is RetailTransaction) { if (TaxSvcHelper.exemptionNumber != String.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared Exemption Number"); TaxSvcHelper.exemptionNumber = string.Empty; } if (TaxSvcHelper.entityUseCode != string.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared EntityUseCode"); TaxSvcHelper.entityUseCode = string.Empty; } if (TaxSvcHelper.retailTransactionTrace != null) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Reset TaxOverride Date"); } } if (posTransaction is RetailTransaction) {</pre>

		<pre> if (TaxSvcHelper.exemptionNumber != string.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared Exemption Number"); //Exemption number scope ends here. TaxSvcHelper.exemptionNumber = string.Empty; } if (TaxSvcHelper.entityUseCode != string.Empty) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Cleared EntityUseCode"); TaxSvcHelper.entityUseCode = string.Empty; } if (TaxSvcHelper.retailTransactionTrace != null) { TaxSvcHelper.retailTransactionTrace.AddToTrace("Reset TaxOverride Date"); } if (posTransaction is CustomerOrderTransaction) { //Question - Do we need to commit this transaction in AvaTax? } } //AVA-</pre>
TransactionTriggers.cproj	TransactionTriggers.cs/PostVoidTransaction	<pre> //AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalc ulation) { string source = "TransactionTriggers.PostVoidTransaction"; string value = "After voiding the transaction..."; LSRetailPosis.ApplicationLog.Log(source, value, LSRetailPosis.LogTraceLevel.Trace); LSRetailPosis.ApplicationLog.WriteAuditEntry(source, value); } //AVA-</pre>

TransactionTriggers.cproj	TransactionTriggers.cs/ PreReturnTransaction	<p>Please search for the following line</p> <pre>LSRetailPosis.ApplicationLog.Log("TransactionTriggers.PreReturnTransaction", "Before returning the transaction...", LSRetailPosis.LogTraceLevel.Trace);</pre> <p>Add the following lines after that</p> <pre>//AVA+ if (!Avalara.Ax.Retail.POSDB.PosConfiguration.RetailDisableTaxCalculation) { try { TaxSvcHelper.isReturnTran = true; if (originalTransaction != null) { if (!string.IsNullOrEmpty(Convert.ToString(((LSRetailPosis.Transaction.PosTransaction)(originalTransaction)).ReceiptId)) { SetAvaAddressForReturnCal(Convert.ToString(((LSRetailPosis.Transaction.PosTransaction)(originalTransaction)).ReceiptId) , Convert.ToString(((LSRetailPosis.Transaction.PosTransaction)(originalTransaction)).StoreId) , Convert.ToString(((LSRetailPosis.Transaction.PosTransaction)(originalTransaction)).TerminalId)); } ReadOnlyCollection<object> rethQObjArr; rethQObjArr = PosApplication.Instance.TransactionServices.InvokeExtension("ava_findallAddresses", ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).ReceiptId, ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).StoreId, ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).TerminalId, ((LSRetailPosis.Transaction.RetailTransaction)(originalTransaction)).TransactionId); if (rethQObjArr.Count > 0) { if (Convert.ToBoolean(rethQObjArr[1])) {</pre>
---------------------------	---	--

		<pre> TaxSvcHelper.OriginalAllLineShippingAddress = Convert.ToString(rethQObjArr[2]); } } } catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n PreReturnTransaction Error : ", ex.Message)); } } //AVA- </pre>
TransactionTrigger s.cproj	TransactionTriggers.c s/	<p>Please go to end of file and search for the 2nd last occurrence of "}" char. And following lines of code . Note : It is nothing but just adding 2 new function in the file</p> <pre> Private void SetAvaAddressForReturnCal(string receiptId, string storeId, string terminalId) { try { TaxSvcHelper.OriginalstoreAddress = null; TaxSvcHelper.OriginalShippingAddress = null; string strconnect = LSRetailPosis.Settings.ApplicationSettings.Database.LocalConnec tion.ConnectionString; System.Data.SqlClient.SqlConnection sqlDataConnection = new SqlConnection(); sqlDataConnection.ConnectionString = strconnect; try { sqlDataConnection.Open(); string sqlString = "select [COMMENT], [transDate] from [dbo].[RETAILTRANSACTIONTABLE] where receiptid= '" + receiptId + "'"; System.Data.SqlClient.SqlCommand sqlCommand = new SqlCommand(sqlString, sqlDataConnection); System.Data.SqlClient.SqlDataReader sqlReader = sqlCommand.ExecuteReader(); string strComment = string.Empty; if (sqlReader.HasRows) { sqlReader.Read(); strComment = sqlReader[0].ToString(); TaxSvcHelper.OriginalDocDate = Convert.ToDateTime(sqlReader[1]); if (!string.IsNullOrEmpty(strComment)) </pre>

		<pre> { Avalara.AvaTax.Adapter.AddressService.Address tmpAddress = TaxSvcHelper.SpiltAddressstring(strComment); if(tmpAddress != null) { if (! (string.IsNullOrEmpty(tmpAddress.Line1)) !(string.IsNullOrEmpty(tmpAddress.PostalCode)) !(string.IsNullOrEmpty(tmpAddress.City)) !(string.IsNullOrEmpty(tmpAddress.Region)) !(string.IsNullOrEmpty(tmpAddress.Country))) { if (TaxSvcHelper.OriginalShippingAddress == null) TaxSvcHelper.OriginalShippingAddress = TaxSvcHelper.CreateAddress(); TaxSvcHelper.OriginalShippingAddress = tmpAddress; } } //Added for return case TaxSvcHelper.exemptionNumber = TaxSvcHelper.SpiltECMSstring(strComment); TaxSvcHelper.entityUseCode = TaxSvcHelper.SpiltEUCodestring(strComment); TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n SetAvaAddressForReturnCal with exemptionNumber = {0}, entityUseCode = {1}", TaxSvcHelper.exemptionNumber, TaxSvcHelper.entityUseCode)); } } catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I n SetAvaAddressForReturnCal-1 Error : ", ex.Message)); } if (TaxSvcHelper.OriginalShippingAddress == null) { AvaSetAddress(receiptId, storeId, terminalId); } if (string.IsNullOrEmpty(TaxSvcHelper.exemptionNumber) string.IsNullOrEmpty(TaxSvcHelper.entityUseCode)) { AvaSetECMSandEUCodeAddress(receiptId, storeId, terminalId); } } catch (Exception ex) { </pre>
--	--	--

```

TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I
n SetAvaAddressForReturnCa1-2 Error : ", ex.Message));
    }
}

    public void AvaSetECMSandEUCodeAddress(string
receiptId, string storeId, string terminalId)
    {
        try
        {
            ReadOnlyCollection<object> rethQObjArr;

            rethQObjArr =
Pos.SystemCore.PosApplication.Instance.TransactionServices.Invo
keExtension("ava_getComment", receiptId, storeId, terminalId);
            if (rethQObjArr.Count > 0)
            {
                if (Convert.ToBoolean(rethQObjArr[1]))
                {

TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("a
va_getComment = ", rethQObjArr[2]));
                    string strComment =
Convert.ToString(rethQObjArr[3]);
                    TaxSvcHelper.exemptionNumber =
TaxSvcHelper.SpiltECMSstring(strComment);
                    TaxSvcHelper.entityUseCode =
TaxSvcHelper.SpiltEUCodestring(strComment);
                    TaxSvcHelper.OriginalDocDate =
Convert.ToDateTime(rethQObjArr[4]);

TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I
n AvaSetECMSandEUCodeAddress with exemptionNumber = {0},
entityUseCode = {1}", TaxSvcHelper.exemptionNumber,
TaxSvcHelper.entityUseCode));
                }
            }
        }
        catch (Exception ex)
        {

TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("E
xception in AvaSetECMSandEUCodeAddress", ex.Message +
Environment.NewLine + ex.StackTrace));
        }

    }

    public void AvaSetAddress(string receiptId, string
storeId, string terminalId)
    {
        try
        {

TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("I
n AvaSetAddress calling ava_getStoreAddress with receiptId =
{0}, storeId = {1}, terminalId = {2}", receiptId, storeId,
terminalId));

```

		<pre>ReadOnlyCollection<object> retHQObjArr; retHQObjArr = Pos.SystemCore.PosApplication.Instance.TransactionServices.InvokeExtension("ava_getStoreAddress", receiptId, storeId, terminalId); // Pos.BlankOperations.BlankOperations.InternalApplication.TransactionServices.InvokeExtension("getStoreAddress", receiptId); if (retHQObjArr.Count > 0) { if (Convert.ToBoolean(retHQObjArr[1])) { if (TaxSvcHelper.OriginalstoreAddress == null) TaxSvcHelper.OriginalstoreAddress = TaxSvcHelper.CreateAddress(); TaxSvcHelper.OriginalstoreAddress = TaxSvcHelper.SpiltAddressstring(Convert.ToString(retHQObjArr[3])); } } TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("ava_getStoreAddress return address = ", retHQObjArr[2])); } } catch (Exception ex) { TaxSvcHelper.retailTransactionTrace.AddToTrace(string.Format("Exception in AvaSetAddress", ex.Message + Environment.NewLine + ex.StackTrace)); } }</pre>
--	--	---